Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 3 (2011) 872–880
Procedia Computer Science
www.elsevier.com/locate/procedia

WCIT 2010

Detecting and investigating crime by means of data mining: a general crime matching framework

MohammadReza Keyvanpour[a] *, Mostafa Javideh[b], Mohammad Reza Ebrahimi[a]

[a]*Department of Computer Engineering, Alzahra University, Tehran, Iran*
[b]*Shamsipoor Technical College, Tehran, Iran*
[a,b] *Islamic Azad University, Qazvin Branch, Qazvin, Iran*

Abstract

Data mining is a way to extract knowledge out of usually large data sets; in other words it is an approach to discover hidden relationships among data by using artificial intelligence methods. The wide range of data mining applications has made it an important field of research. Criminology is one of the most important fields for applying data mining. Criminology is a process that aims to identify crime characteristics. Actually crime analysis includes exploring and detecting crimes and their relationships with criminals. The high volume of crime datasets and also the complexity of relationships between these kinds of data have made criminology an appropriate field for applying data mining techniques. Identifying crime characteristics is the first step for developing further analysis. The knowledge that is gained from data mining approaches is a very useful tool which can help and support police forces. An approach based on data mining techniques is discussed in this paper to extract important entities from police narrative reports which are written in plain text. By using this approach, crime data can be automatically entered into a database, in law enforcement agencies. We have also applied a SOM clustering method in the scope of crime analysis and finally we will use the clustering results in order to perform crime matching process.

Keywords: Data Mining; Crime Analysis; Crime Investigation; Criminology; Neural Network; Text Mining

1. Introduction

Undoubtedly, the circumstances of humans' social life, makes it vital to encounter a phenomenon known as crime. So we always need to the knowledge of crime analysis as an efficient combating tool. Crime analysis basically includes leveraging a systematic approach for identifying, discovering and sometimes predicting crime incidents. The input of a crime analysis system is consisted of data and information assigned to crime variables and the output includes the answer to investigative and analytical questions, knowledge extraction and finally visualization of the results. The complex nature of crime and criminality-related data and also the existence of hidden and maybe intangible relations between them have made data mining a rapidly growing field among criminologists, crime investigators and crime analysts. The large volumes of crime-related data existed in police departments and also the complexity of relationships between these kinds of data has forced the traditional crime analysis methods to become obsolete. These methods require a considerable amount of time and human resources on

* Mohammad Reza Keyvanpour.
E-mail address: keyvanpour@alzahra.ac.ir

1877-0509
doi:10.1016/j.procs.2010.12.143

one hand and on the other hand they are not able to get all effective parameters/relationships involved due to their high amount of human interference. These deficiencies have revealed the necessity of using a systematic and intelligent approach for crime investigation more than ever. Data mining techniques can be the key solution.

Considering the novelty of leveraging data mining techniques in the domain of crime analysis, more researches seem to be required in this field. Some of the key concepts of intelligent crime analysis and crime data mining techniques are discussed in the next sections. These techniques include data clustering and artificial neural networks. The former helps to categorize crimes unsupervisely and the later can be used for crime pattern recognition. Finally a proposed method for crime matching purposes has been covered.

This article is consisted of 5 main sections. The second section has been devoted to a brief survey on related researches and existing intelligent crime analysis software utilities, which are all common in the issue of applying data mining techniques in the domain of crime investigation and analysis. In section 3, the basic components of crime analysis including the concepts of crime variables and crime matching have been covered. Section 4 presents a systematic approach for crime matching by means of common clustering methods, SOM and MLP neural networks. Finally, section 5 is devoted to conclusion and future works of the authors.

2. Related Works

In the recent decade, a great deal of scientific researches and studies have been performed on crime data mining. The results are usually emerged in the aspect of new software applications for detecting and analyzing crime data. In [1] the authors introduce a general overview on applying intelligent crime analysis methods including neural networks, Bayesian networks, and genetic algorithms in predicting and matching crime incidents. In [2], neural networks have been applied for crime data clustering and crime data classification through using both supervised and unsupervised learning methods. The COPLINK national project [3-6] which was originally developed by the University of Arizona Artificial Intelligence Lab with funding from the National Institute of Justice represents a prominent framework for text mining, classification and clustering of crime data aiming to accomplish relatively complex crime analysis. The project contains two fundamental components: 1) COPLINK CONNECT and 2) COPLINK DETECT. The former handles data preprocessing and data gathering burdens and the later deals with extracting patterns out of large volumes of crime data by using data mining and artificial intelligence.

3. An Introduction to Intelligent Crime Analysis' fundamentals

Crime variables and crime matching are two main components which are usually involved in crime analysis process. Crime variables are so important because the analysis algorithm initially works on them. On the other hand, the importance of crime matching is due to its vast usage in intelligent crime detection. These two subjects are covered in this section as intelligent crime analysis fundamentals.

3.1. Crime Variables

There are some parameters which can describe crime characteristics somehow uniquely. These unique crime parameters known as crime variables are the main subject of crime analysis process. Regardless of crime type, we can categorize different types of crime variables into three general groups:

1) Spatio-temporal crime variables (e.g. crime location coordinates or the time of occurrence)
2) Crime natural specifications (e.g. crime scene characteristics, offender's behavioral pattern)
3) Offender profiles (e.g. offender specifications (age, sex, race, etc.))

It is notable that every crime type includes its own specific crime variables. As an example, the crime variables for homicide will not be the same as the crime variables for larceny.

Table 1. crime variables for Burglary Dwelling Houses (BDH)

Categories	Related variables
location of entry	Walls, roof, window, etc.

method of entry	Climbing, drilling, destroying, breaking, tunnel, etc.
type of dwelling house	Apartment, villa, bungalow, etc.
type of searching	tidy, untidy, all rooms, just one place, etc.
Location of exit	Walls, roof, window, etc.
methods of offender interaction with the environment	Lock the door after entering, manipulate the alarm, killing the watchdog, etc.

Even in the scope of larceny crimes, there are various kinds of crime variables for different types of larceny like burglary, robbery, auto-theft, and so on. Therefore, different types of crimes require analyzing different kinds of crime variables. Table 1, lists some of the most important variables for Burglary from dwelling Houses (BDH).

3.2. Crime Matching

The process of assigning crimes or criminals to the previous solved or unsolved crime incidents is known as crime matching. In fact, using crime matching in forensic investigations has two aspects:
1) Assuming that one or more offenders responsible for a specific crime have been arrested. Crime matching process is used for assigning the previous unsolved crimes to the arrested offenders.
2) Considering a situation that police has been warned about a new unsolved crime incident so corresponding criminals are not discovered yet. In such a situation, crime matching is applied to suggest some prolific offenders as probable suspects based on the offender profiles and the method of committing criminal act.

Most of the offenders commit several crimes before they get arrested. When a criminal is found to be responsible for a specific crime instance, investigators will usually prepare a list of similar crimes by running some SQL queries on previous crime data in order to match instances. This traditional method of crime matching suffers from two main drawbacks: 1) There is no general rule for determining WHERE condition in the SQL query statement so it is required to perform the time-consuming task of testing multiple queries with different WHERE clauses. 2) Because of the natural simplicity of the queries, the result list lacks enough accuracy. In fact, crime matching process includes 3 main phases:
1) Feature selection: includes extracting crime features which are effective enough to be involved in analysis process. In the domain of intelligent crime analysis, features are a subset of crime variables (table 1). Genetic algorithm, Best-First and Forward Selection algorithms considered as efficient methods of feature selection [1].
2) Encoding: Selected features should be encoded in an appropriate way to perform the proper algorithms.
3) Matching algorithm: The algorithms determine groups of crimes or offenders most similar to each other. Each algorithm works based on its own similarity measures. K-Nearest Neighbor (KNN), Tversky's Contrast Model and different clustering algorithms can be utilized as matching algorithm [1].

In this article, a crime matching approach has been presented by utilizing Artificial Neural Network (ANN) and binary encoding. The approach aims to improve the accuracy and reliability of intelligent crime detection process.

4. Proposed Method's Components: Toward a Crime Matching Framework

The proposed method has been represented with the purpose of utilizing data mining techniques in the domain of intelligent crime analysis and it includes a systematic approach for leveraging three types of data mining techniques as its basic components: 1) Entity Extraction as a branch of text mining, 2) crime data clustering and 3) Neural Network as an engine for crime matching process.

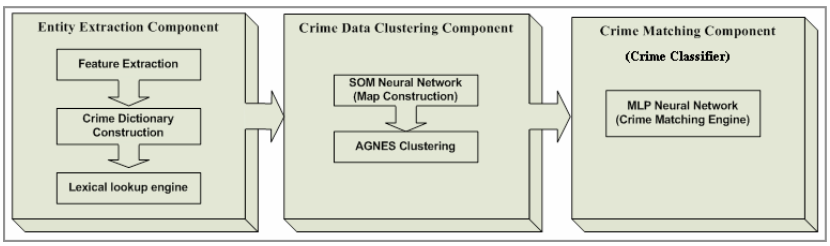

Fig. 1. the proposed method's components

These techniques will be discussed and dissected in this section. Figure (1) shows the relationships between these components. It is also illustrates the inner sub-components.

4.1. Automatic Entity Extraction from Crime Narrative Reports: the First Component

Textual crime scene reports and final narrative reports which usually exist in the data bases of crime-related organizations, are considered as the richest source of intelligent analysis, but unfortunately, these useful sources for crime data mining have no prespecified structure! In other words they are written in plain text and as a result, it is a super-challenging task to extract desired information out of non-structured texts. So leveraging intelligent techniques of text mining will be efficient for achieving this goal. One of the most common techniques in text mining is called Named Entity Extraction which is considered as a branch of information extraction. Named Entity Extraction aims to identify desired entities' value out of a plain text [7]. Undoubtedly, in the domain of intelligent crime detection, these entities are crime variables. Crime entity extraction can be regarded as a data preprocess step.

Common methods for extracting entities out of narrative reports can be categorized as 4 kinds of groups: 1) Lexical Lookup, 2) Rule-based, 3) Static-based and 4) Machine Learning-based methods. A lexical lookup approach has been used for crime entity extraction in this research. The approach constructed a lookup table which contains the aggregate interested words and expressions chosen by a crime domain expert. We named this lookup table as crime dictionary. Then, a simple search engine was developed which was able to find the dictionary's words among the textual data of police narrative reports. We found this automatic approach more efficient than reading the narrative reports manually and entering them in a structured database which is really a time-consuming task. Using such an approach, we were also able to refine the output of the search engine by leveraging general words such as "Mr.", "Mrs.", "organization" and etc.

In the process of entity extraction we faced some challenging problems which are summarized as follows:

1) Extracting crime-related entities out of police narrative reports is naturally harder than extracting entities out of other kinds of texts in other problem domains. The reason is that in spite of common entities such as persons' names, organizations' names and places, which are usually seen in non-crime-related texts, there are a wider variety of entities such as narcotics names, crime location coordinates, vehicle models and numbers involved in the scope of criminology.

2) Misspelling and grammatical mistakes are extremely common in police narrative reports. These problems make the process of entity extraction out of crime-related data much more complex in comparison with other reports like news texts.

4.2. Crime Data Clustering

In this section, considerations and challenges of leveraging classic hierarchical and partitioning clustering techniques in intelligent crime analysis has been discussed. Subsequently, a proposed approach for crime data clustering is represented which utilizes SOM neural network in order to overcome other clustering techniques drawbacks.

The binary nature of crime behavioral variables may be regarded as a challenge, because committing clustering and analysis process on these types of variables requires some elegance. The binary encoding causes the popular Euclidian distance measure- which is commonly used for continuous types of variables- to be useless. The reason is that behaving binary quantities like continuous quantities can lead to misleading results in clustering process [8]. Some other distance functions should be leveraged which are specific for achieving the similarity between binary data objects. These functions calculate the dissimilarity between two objects as their corresponding distance. The distance (dissimilarity) between two objects can be calculated by equation (1).

$$D(i, j) = 1 - S(i, j) \quad (1)$$

Where S and D correspondingly represent functions of similarity and dissimilarity between two binary sequences i and j. In order to calculate the similarity, assuming that both sequences are equal in length, we need to define

variables *a*, *b*, *c* and *d* as follows:

a represents the number of corresponding bits with value 1 in both bit sequences.

b represents the number of bits which equal 1 in the first bit sequence but equal 0 in the second bit sequence.

c represents the number of bits which equal 0 in the first bit sequence but equal 1 in the second bit sequence.

Finally, *d* represents the number of corresponding bits with value 0 in both bit sequences.

Table 2. Different modes of bit comparison

Respective variable	Bit value in the first sequence	Bit value in the second sequence
a	1	1
b	1	0
c	0	1
d	0	0

According to the above definitions the similarity between bit sequences ***i*** and ***j*** can be measured by the following equations [9]:

- Simple Match Coefficient: $s(i,j) = (p+s) / (p+q+r+s)$
- Rao's Coefficient: $s(i,j) = p / (p+q+r+s)$
- Jaccard Coefficient: $s(i,j) = p / (p+q+r)$

It is also notable that popular classic κ-means algorithm cannot be used for clustering binary data objects. The reason is that, classic κ-means' calculated centroids are not binary. To overcome the problem, using κ-medoids partitioning clustering method is suggested, in which one of the binary objects in a cluster represents the center of that cluster. Complementarily, if hierarchical clustering method is chosen in order to cluster binary data objects, it will be impossible to use *Average-link* or *centrroid distance* [10] methods as inter-cluster distance measuring strategy. Instead, we can use *single-link* or *complete-link* distance measures [10]. It is also worthy to know that because different types of hierarchical clustering methods have time complexity of $O(n^2)$ *and* $O(n^2 logn)$[11], it is not lucrative to use hierarchical clustering with large amount of high-dimensional crime data.

4.3. Self-Organizing Map Neural Network: the Second Component

As already mentioned, high number of dimensions of crime-related data is a challenge that affects the approach which might be chosen for crime clustering. The approach should be able to deal with many various crime variables. The SOM's ability to map the high-dimensional data spaces into low-dimensional views is the key solution to deal with this challenge. SOM neural network reduces the number of dimensions, while preserving the *data topology* [8, 12]. So statistical distribution of the network's output will be a low-dimensional appearance of the high-dimensional data which are fed into the network as its input. This is considered as an elegant way for high-dimensional data visualization. Also There are some other useful visualization algorithms such as *U-Matrix* [13] and Component Planes [14,15] which are not the subject of this paper. Moreover, one of the most significant advantages of using SOM neural networks is their natural tendency to be leveraged in parallel processing and distributed architectures. As a result, the method seems to be efficient in facing with large volumes of crime-related data. It is also applicable in dealing with data with non-linear statistical distributions [8]. Consequently, due to the non-linear distribution of criminal data it is suggested to exploit SOM abilities in this field.

The proposed method of crime data clustering which was used in this research included a two-step approach; in the first step a self organizing neural network was used to extract the *feature map*, subsequently, κ-means popular clustering algorithm categorized the network output as the second step. *But what was the exploited neural network architecture?* The input layer was consisted of 21 crime variables encoded as binary digits which were obtained by feature extraction process. The output layer was consisted of a 2-D 25×25 array of neurons. Each neuron in the output layer was related to all of the neurons in the input layer through weighted connections. The SOM neural

network algorithm was initialized by random connection weights. We trained the network by sending some predetermined binary individuals into the input layer. In each iteration of the training process the weights are readjusted. Eventually, after the training process ends, the weights of the output layer mimic the real high-dimensional data distribution.

How does the training process works? Feeding the train data into the network, the network computes the similarity between the input data and all of the neurons of the output layer. An output layer neuron which is the most similar one to the input data is determined as the *winner neuron*. Sending the input data into the network, the distance of the output layer neurons and the input data is computed. Subsequently, the winner's weight and also some of its neighbors are readjusted according to the value of *neighborhood function*. It is worthy to note that the value of $\Theta(v)$ will be decreased gradually through the training process till it is limited only to the winner neuron. Equation (5) shows how the weights are readjusted in a SOM neural network.

$$Wv^{+} = Wv^{-} + \Theta(v) \times \alpha \times (X - Wv^{-}) \quad (5)$$

In which W_v^- represents the current weight for the neuron v, $\Theta(v)$ is the neighborhood function which was discussed above, X represents the input data and α is the *learning coefficient* which is decreased in value through training process and W_v^+ is the readjusted weight of neuron v that assigned to the corresponding neuron. When the training step is finished and the data map is generated by the SOM neural network, the second stage starts. This step performs a partitional (e.g. k-means) or a hierarchical (e.g. AGNES or DIANA) clustering algorithm on the SOM generated map. It means that neurons of the output layer (SOM lattice) which have the closest weights to each other are grouped as a cluster. These clusters are used for further analysis (i.e. crime matching which is discussed in the next section). We used the AGNES algorithm [8] for the second stage, which is a hierarchical bottom-up clustering algorithm. AGNES was chosen because it's a simple but an accurate method. It should be minded that AGNES suffers from high time and space complexity ($O(n3)$). To alleviate this problem, it is possible to use k-means algorithm which has a very low time complexity ($O(n)$). In this way, we also have the choice to exploit the SOM's output (graphical map) for estimating the optimum *number of clusters* and also the *initialization seeds* in order to avoid random initialization in k-means algorithm.

4.4. Proposed Crime Matching engine: Crime Classifier Component

Undoubtedly, crime matching is one of the most important requirements in analyzing and investigating crimes. It aims to match crimes to criminals and vice versa. As already mentioned, there are 2 practical modes for crime matching: 1. A new crime incident is occurred and the goal is to identify the probable offenders responsible for the corresponding crime incident, and 2. an offender is arrested and the investigative aims to assign some unsolved similar crime instances to the current offender. This research mainly focuses on using crime matching with behavioral burglary crime variables. Burglary crime variables were categorized into 4 groups: 1. Burglary location types, 2. the ways that the offender interacts with crime environment, 3. types of entry method and 4. the tools and instruments that the offender has used. For the sake of clarity, table (3) shows a condensed list of crime variables for type of entry method which are binary encoded.

Table (3). modus operandi for entry methods in a specific crime instance

Entry Methods	Binary encoded value
Front Door	0
Rear Door	0
Window	1
Breaking the Window	1
Climb	1
Damage Locks	0
Terrace	1

Neighbor's Houses	0

In fact, the method which the offender has used to enter the premise is encoded in the form of a relatively unique binary modus operandi for a specific burglary incident. As an example, the binary string respective to the entry method in the burglary instance shown in table (3) is "00111010". This way of encoding, provides the crime matching utilization. In order to match crimes in the first application mode (i.e. after detecting the new unsolved burglary), the investigative usually prepare a narrative report for the crime, and the report will be fed into the automatic entity extraction component for extracting the respective burglary incident's features. Subsequently, the encoded binary string of mentioned crime will be generated and finally the crime matching engine calculates the similarity of the new unsolved crime to the clustered solved burglary instances (clustering engine was discussed in section 4-2 and 4-3.). Based on the similarity of the new crime to the crimes resident in determined clusters, the respective probable offenders will be identified. Contrarily, in order to match crimes in the second application mode (i.e. when an offender is arrested for a specific burglary), we will send the bit strings of all of his/her burglaries to the crime matching engine in order to find the similar unsolved burglaries to the offender's recent known burglaries. This way, we will be able to probably assign the unsolved burglaries to the arrested offenders. These kinds of information can help investigative to have more efficient interviews with the arrested offender.

But how does the crime matching engine work? It uses a *classification* process by means of a Multi-Layer Perceptron (MLP) neural network with *back-propagation* training method [16]. One of the most significant benefits of leveraging an MLP classifier is its high ability to tolerate noisy data instances beside the ability to be used in parallel data processing. We devise a MLP neural network for each category of burglary crime variables including burglary location types, the type of the offender interactions with crime environment, types of entry method and the tools used by the offender. The topology of MLP contains 3 layers of neurons (i.e. input output and one hidden layer). It can be proved that having just one hidden layer in the neural network's architecture is sufficient to approximate any linear or non-linear function [17]. The number of neurons in input layer was designed to be equal to the number of crime variables involved in analysis. For example, the devised neural network for types of entry method should contain 8 neurons in the input layer (one neuron for every feature showed in table (3)). The number of output layer neurons was devised to be equal to the number of clusters which were determined by crime data clustering component (section 4-3). It is notable that there is no general rule for estimate the optimum number of neurons of the hidden layer of the network in design time, but it is somehow practically proved in [18] that the MLP networks will be more efficient if the number of the hidden layer neurons is chosen according to equation (6).

$$m = \alpha \times \sqrt{n_p \times n_a} \qquad (6)$$

Where m is the suggested number of hidden layer neurons, np and na represent the number of input and output layer neurons respectively and finally α is a coefficient which was set to be 4 based on some try-and-error experiments. Learning process commences with random assignment of weights to the connections between neurons. Subsequently, the train input vectors (input data which we already know what cluster they belong to) are sent to the network. A target bit string introduce to the network for each train input data. This target string was set to have 0 for all of its bit values except the one that represents the respective cluster. For example, assume a network that has just 3 neurons in its output layer (i.e. 3 cluster has been obtained in the clustering phase which was discussed in section 4-3). Also assume that a train input vector belongs to cluster number 2, So the target bit string will be "010".

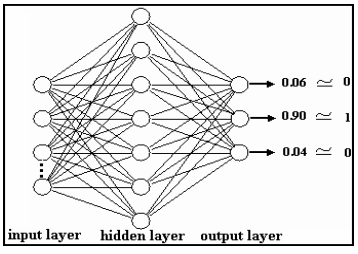

Fig. 2. Interpreting the MLP's output

Anyway, the network attempts to readjust the weights according to the given input-target pairs. After the training process, the network will be able to classify the test data and generates a bit sequence as its output (e.g. "010"). Figure (2) shows how did we interpret the MLP's generated output as a bit string.

5. Conclusion and Future Works

In this research some of the most significant capabilities of data mining techniques were leveraged through a multi-purpose framework for intelligent crime investigation. The framework exploited a systematic approach for using SOM and MLP neural networks for clustering and classifying crime data. Considerations and challenges of using hierarchical/partitional clustering techniques in crime data clustering were also discussed. We hope we can make crime spatio-temporal data involved in the analysis besides behavioral crime variables. We also intend to implement this framework as an integrated enterprise software.

References

1. G.C. Oatley, J. Zeleznikow, B.W. Ewart, "*Matching and Predicting Crimes,"* In Applications and Innovations in Intelligent Systems XII in Proceedings of AI2004, The Twenty-fourth SGAI International Conference on Knowledge Based Systems and Applications of Artificial Intelligence. Ann Macintosh, Richard Ellis and Tony Allen Ed. London: Springer, , pp. 19-32, 2004.
2. R. William Adderley, "*The use of data mining techniques in crime trend analysis and offender profiling*," Ph.D. thesis, University of Wolverhampton, Wolverhampton, England , 2007.
3. Y. Xiang, M. Chau, H. Atabakhsh, H. Chen, *"Visualizing criminal relationships: comparison of a hyperbolic tree and a hierarchical list,*" Decision support systems, Elsevier Science Publishers, vol. 41 no.1, pp: 69-83, Nov. 2005.
4. . Chen, W. Chung, Y. Qin, M. Chau, J. Xu, G. Wang, R. Zheng, and H. Atabakhsh, "Crime Data Mining: An Overview and Case Studies," in Proceedings of the 3rd National Conference for Digital Government Research (dg.o 2003), pp. 1-5, Boston, MA, May 18-21, 2003.
5. H. Chen, H. Atabakhsh, T. Petersen, J. Schroeder, T. Buetow, L. Chaboya, C. O'Toole, M. Chau, T. Cushna, D. Casey, Z. Huang, *"COPLINK: Visualization for Crime Analysis,"* ACM International Conference Proceeding Series, Proceedings of the 2003 annual national conference on digital government research, , Vol. 130, pp 1-6, Boston, MA, 2003.
6. R.V. Hauck, H. Atabakhsh, P. Ongvasith, H. Gupta, H. Chen, *"Using Coplink to Analyze Criminal-Justice Data,"* Computer, vol. 35, no. 3, pp. 30-37, Mar. 2002.
7. H. Chen, W. Chung, J.J Xu, G. Wang, Y. Qin and M. Chau, "Crime Data Mining: A General Framework and Some Examples," *Computer*, vol. 37, no. 4, pp. 50-56, Apr. 2004.
8. J. Han and M. Kamber, *Data Mining Concepts and Techniques*, 2nd ed. Morgan Kaufmann, Nov. 3, 2005.
9. D. J. Hand, H. Mannila, P. Smyth, Principles of Data Mining, Massachusetts, MIT Press, 2001.
10. G.K. Gupta, *Introduction to Data Mining with Case Studies,* New Delhi: Prentice-hall of India, 2006.
11. Y.J. Oyang, C.Y. Chen , T.W. Yang, "A Study on the Hierarchical Data Clustering Algorithm Based on Gravity Theory," Proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery, p.350-361, Sep. 03-05, 2001.
12. J. Huysmans, D. Martens, B. Baesens, J. Vanthienen and T. Van Gestel, *"Country Corruption Analysis with Self Organizing Maps and Support Vector Machines,"* Proceedings of the Tenth Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2006), Workshop on Intelligence and Security Informatics (WISI), vol. 3917, pp. 103-114, Singapore: Springer-Verlag, 2006.
13. M. Aurélio S. da Silva, A. M. V. Monteiro and J. S. Medeiros, "*Visualization of Geospatial data by component planes and U-Matrix,"* 6th Brazilian Symposium on Geoinformatics, São Paulo, Brazil, 22-24 Nov. 2004.
14. R. Neumayer, R. Mayer, G. Pölzlbauer and A. Rauber, "*The Metro Visualisation of Component Planes for Self-Organising Maps,"* In Proceedings of the 20th International Joint Conference on Neural Networks (IJCNN'07), Orlando, FL, USA: IEEE Computer Society, Aug. 12 – 17, 2007, pp. 2788-2793.

15. G. Pölzlbauer, M. Dittenbach, A. Rauber, "*Gradient visualization of grouped component planes on the SOM lattice,"* In Proceedings of the 5th Workshop On Self-Organizing Maps Paris (WSOM 2005), Sep. 5-8 2005, Paris, France, pp. 331-338.
16. K. l. Priddy, p. e. Keller, *Artificial Neural Networks*, New Delhi: Prentice-hall of India, 2007.
17. K. Hornik,*" Approximation capabilities of multi-layer feedforward networks," Neural Networks*, vol. 4, no. 2, pp. 251–257. Oxford, UK, Elsevier Science Ltd. 1991.
18. Z. Tang and J. MacLennan, *Data Mining with SQL Server 2005*, Indianapolis, USA: Wiley Publishing, 2005.